

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2003-195749

(P2003-195749A)

(43) 公開日 平成15年7月9日 (2003.7.9)

(51) Int.Cl.<sup>7</sup>

G 0 9 C 1/00

識別記号

6 1 0

F I

G 0 9 C 1/00

テーマコード(参考)

6 1 0 A 5 J 1 0 4

審査請求 未請求 請求項の数 9 O L (全 18 頁)

(21) 出願番号 特願2001-394109(P2001-394109)

(22) 出願日 平成13年12月26日 (2001. 12. 26)

特許法第30条第1項適用申請有り 平成13年9月26日～  
28日 社団法人情報処理学会開催の「第63回 (平成13年  
後期) 全国大会」において文書をもって発表

(71) 出願人 396021944

株式会社デンソークリエイト

愛知県名古屋市中区錦二丁目15番20号

(71) 出願人 301080002

神保 雅一

岐阜県各務原市緑苑南4-112

(72) 発明者 陳 志松

愛知県名古屋市中区錦二丁目15番20号 株  
式会社デンソークリエイト内

(74) 代理人 100082500

弁理士 足立 勉

最終頁に続く

(54) 【発明の名称】 データ変換装置、データ変換プログラム、記録媒体及びデータ変換方法

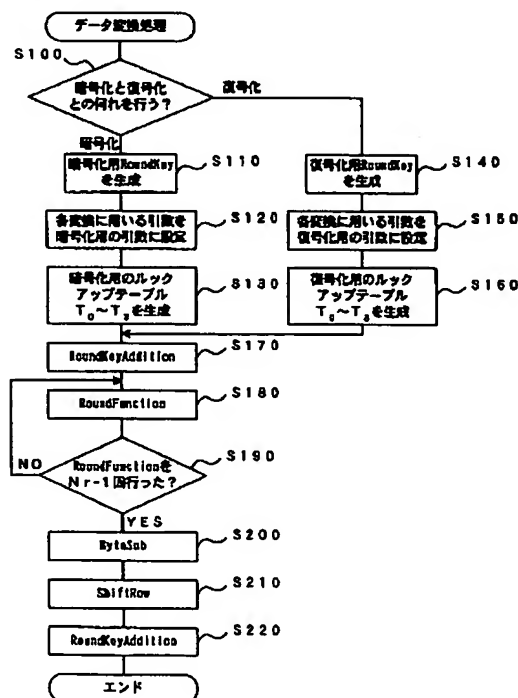
(57) 【要約】

【課題】 AES暗号方式を用いてデータを暗号化又は復号化する装置に必要なデータ記憶容量を小さくする。

【解決手段】 このデータ変換処理では、AES暗号方式でデータを暗号化又は復号化するために行うByte Sub、ShiftRow、MixColumn及びRoundKey Additionの4種類の変換について、各変換で用いる指数を、暗号化と復号化とで異なる所定の指数に設定することで (S100～S160)、データの暗号化及び復号化を、AES暗号方式の暗号化手順と同じ順序で上記4種類の変換を施すことにより行うようになっている (170～S220)。また、S180では、4行4列の行列で表される128ビットの処理対象データに上記4種類の変換を順に施すことで得られる処理後データを、下記の式により求めるようになっている。

【数1】

$$\begin{aligned} & \text{処理後データ} = \text{処理前データ} \times \text{変換行列} \\ & \text{変換行列} = \begin{pmatrix} T_0 & T_1 & T_2 & T_3 \\ T_4 & T_5 & T_6 & T_7 \\ T_8 & T_9 & T_{10} & T_{11} \\ T_{12} & T_{13} & T_{14} & T_{15} \end{pmatrix} \end{aligned}$$



BEST AVAILABLE COPY

【特許請求の範囲】

【請求項1】 4行4列の行列で表される128ビット  
の変換対象データに、AES暗号方式におけるByte  
Sub、ShiftRow及びMixColumnの各

変換を順に施すことで得られる変換後データを、下記式  
(1)により求めるように構成されていることを特徴と  
するデータ変換装置。

【数1】

$$h_j = T_0[a_{0,j}] \oplus T_1[a_{1,j-c_1}] \\ \oplus T_2[a_{2,j-c_2}] \oplus T_3[a_{3,j-c_3}] \cdots \text{式(1)}$$

$$\left( \begin{array}{l} \text{但し、} h_j : \text{変換後データにおける} j \text{列目} (j=0, 1, 2, 3) \\ \text{のデータ(32ビット)} \\ T_0 \sim T_3 : 8 \text{ビットのデータを32ビットのデータ} \\ \text{に変換するルックアップテーブル} \\ a_{i,j} : \text{変換対象データにおける} i \text{行目} (i=0, 1, 2, 3) \\ \text{の} j \text{列目のデータ(8ビット)} \\ \text{(尚、} a_{i,j-c_i} \text{は、ShiftRowの変換を施す} \\ \text{ことにより} i \text{行目の} j \text{列目へ移動するデータ} \\ \text{を表す)} \\ \oplus : \text{排他論理和} \end{array} \right)$$

【請求項2】 4行4列の行列で表される128ビット  
の変換対象データに、AES暗号方式におけるByte  
Sub、ShiftRow及びMixColumnの各  
変換を順に施すことで得られる変換後データを、下記式

20 (1)により求める機能をコンピュータに実現させるた  
めのデータ変換プログラム。

【数2】

$$h_j = T_0[a_{0,j}] \oplus T_1[a_{1,j-c_1}] \\ \oplus T_2[a_{2,j-c_2}] \oplus T_3[a_{3,j-c_3}] \cdots \text{式(1)}$$

$$\left( \begin{array}{l} \text{但し、} h_j : \text{変換後データにおける} j \text{列目} (j=0, 1, 2, 3) \\ \text{のデータ(32ビット)} \\ T_0 \sim T_3 : 8 \text{ビットのデータを32ビットのデータ} \\ \text{に変換するルックアップテーブル} \\ a_{i,j} : \text{変換対象データにおける} i \text{行目} (i=0, 1, 2, 3) \\ \text{の} j \text{列目のデータ(8ビット)} \\ \text{(尚、} a_{i,j-c_i} \text{は、ShiftRowの変換を施す} \\ \text{ことにより} i \text{行目の} j \text{列目へ移動するデータ} \\ \text{を表す)} \\ \oplus : \text{排他論理和} \end{array} \right)$$

【請求項3】 請求項2に記載のデータ変換プログラム  
が記録されたコンピュータ読み取り可能な記録媒体。

【請求項4】 4行4列の行列で表される128ビット  
の変換対象データに、AES暗号方式におけるByte

Sub、ShiftRow及びMixColumnの各  
40 変換を順に施すことで得られる変換後データを、下記式  
(1)により求めることを特徴とするデータ変換方法。  
【数3】

$$h_j = T_0[a_{0,j}] \oplus T_1[a_{1,j-c_1}] \oplus T_2[a_{2,j-c_2}] \oplus T_3[a_{3,j-c_3}] \dots \text{式(1)}$$

$$\left( \begin{array}{l} \text{但し、} h_j : \text{変換後データにおける} j \text{列目} (j=0, 1, 2, 3) \\ \text{のデータ(32ビット)} \\ T_0 \sim T_3 : 8 \text{ビットのデータを32ビットのデータ} \\ \text{に変換するルックアップテーブル} \\ a_{i,j} : \text{変換対象データにおける} i \text{行目} (i=0, 1, 2, 3) \\ \text{の} j \text{列目のデータ(8ビット)} \\ \text{(尚、} a_{i,j-c_i} \text{は、ShiftRowの変換を施す} \\ \text{ことにより} i \text{行目の} j \text{列目へ移動するデータ} \\ \text{を表す)} \\ \oplus : \text{排他論理和} \end{array} \right)$$

【請求項5】 AES暗号方式を用いて暗号化されたデータを、下記(a1)～(a4)の条件に従い復号化することを特徴とするデータ変換方法。

(a1) : ByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換を、AES暗号方式の暗号化手順と同じ順序で行う。

(a2) : 上記(a1)に従い行うByteSub、ShiftRow及びMixColumnでは、暗号化で行われる変換の逆変換となるような引数を用いる。

(a3) : 上記(a1)に従い行うRoundKeyAdditionでは、暗号化で用いられた複数のRoundKeyを、暗号化と逆の順序で用いる。

(a4) : 更に、上記(a1)に従い行うRoundK

eyAdditionのうち、MixColumnの次に行うRoundKeyAdditionでは、上記

(a3)に従い用いるRoundKeyに、復号化で行うMixColumnの変換を施したものを、RoundKeyとして用いる。

【請求項6】 請求項5に記載のデータ変換方法において、

4行4列の行列で表される128ビットの変換対象データに、ByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られる変換後データを、下記式(1)により求めることを特徴とするデータ変換方法。

【数4】

$$h_j = T_0[a_{0,j}] \oplus T_1[a_{1,j-c_1}] \oplus T_2[a_{2,j-c_2}] \oplus T_3[a_{3,j-c_3}] \dots \text{式(1)}$$

$$\left( \begin{array}{l} \text{但し、} h_j : \text{変換後データにおける} j \text{列目} (j=0, 1, 2, 3) \\ \text{のデータ(32ビット)} \\ T_0 \sim T_3 : 8 \text{ビットのデータを32ビットのデータ} \\ \text{に変換するルックアップテーブル} \\ a_{i,j} : \text{変換対象データにおける} i \text{行目} (i=0, 1, 2, 3) \\ \text{の} j \text{列目のデータ(8ビット)} \\ \text{(尚、} a_{i,j-c_i} \text{は、ShiftRowの変換を施す} \\ \text{ことにより} i \text{行目の} j \text{列目へ移動するデータ} \\ \text{を表す)} \\ \oplus : \text{排他論理和} \end{array} \right)$$

【請求項7】 AES暗号方式を用いたデータの暗号化と、AES暗号方式を用いて暗号化されたデータの復号化とを、コンピュータに行わせるためのデータ変換プログラムであって、

データの暗号化及び復号化で共通に用いられると共に、該データに対してByteSub、ShiftRow、

MixColumn及びRoundKeyAdditionの各変換をAES暗号方式の暗号化手順と同じ順序で行うための共通プログラムと、

前記共通プログラムでの各変換に用いられる引数を、暗号化の場合にはAES暗号方式の暗号化手順において用いられる引数に設定し、復号化の場合には下記(b1)

## 5

～(b3)の条件に従い設定する設定プログラムと、  
を備えたことを特徴とするデータ変換プログラム。

(b1): ByteSub、ShiftRow及びMixColumnでは、暗号化で行われる変換の逆変換となるような引数が用いられるようにする。

(b2): RoundKeyAdditionで用いられる引数であるRoundKeyとしては、暗号化で用いられた複数のRoundKeyが、暗号化と逆の順序で用いられるようにする。

(b3): 更に、MixColumnの次に行われるRoundKeyAdditionで用いられる引数であるRoundKeyとしては、上記(b2)に従い

$$h_j = T_0[a_{0,j}] \oplus T_1[a_{1,j-c1}] \oplus T_2[a_{2,j-c2}] \oplus T_3[a_{3,j-c3}] \cdots \text{式(1)}$$

$$\left( \begin{array}{l} \text{但し、} h_j : \text{変換後データにおける} j \text{列目} (j=0, 1, 2, 3) \\ \text{のデータ(32ビット)} \\ T_0 \sim T_3 : 8 \text{ビットのデータを32ビットのデータ} \\ \text{に変換するルックアップテーブル} \\ a_{i,j} : \text{変換対象データにおける} i \text{行目} (i=0, 1, 2, 3) \\ \text{の} j \text{列目のデータ(8ビット)} \\ \text{(尚、} a_{i,j-ci} \text{は、ShiftRowの変換を施す} \\ \text{ことにより} i \text{行目の} j \text{列目へ移動するデータ} \\ \text{を表す)} \\ \oplus : \text{排他論理和} \end{array} \right)$$

【請求項9】 請求項7又は請求項8に記載のデータ変換プログラムが記録されたコンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、AES暗号方式を用いてデータを暗号化又は復号化する技術に関する。

【0002】

【従来の技術】従来より、データを暗号化するための様々な暗号方式が知られている。そして、最近では、米国連邦標準技術局(NIST)により、AES(Advanced Encryption Standard)と呼ばれる新しい共通暗号方式が発表され、今後の普及が見込まれている。

【0003】ここで、周知ではあるが、AES暗号方式(以下、単に「AES」ともいう)の概要について説明する。AESは、図7に示すように、ブロック長が128ビットのブロック暗号である。そして、この128ビットのブロックデータは、1バイト(8ビット)単位のデータ(以下、「部分データ」ともいう)に分割され、 $4 \times 4$ (4行4列)の正方形行列として表現される。尚、図7では、ブロックデータにおける*i*行目( $i=0, 1, 2, 3$ )の*j*列目( $j=0, 1, 2, 3$ )の部分デ

## 6

られるRoundKeyに、復号化で行われるMixColumnの変換を施したものが用いられるようにする。

【請求項8】 請求項7に記載のデータ変換プログラムにおいて、

前記共通プログラムでは、4行4列の行列で表される128ビットの変換対象データに、ByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られる変換後データを、下記式(1)により求めるようになっていることを特徴とするデータ変換プログラム。

【数5】

ータを、 $b(i, j)$ と表している。

【0004】また、図8に示すように、AESに用いられる鍵の鍵長は128ビット、192ビット、256ビットの3種類である。そして、この鍵も、1バイト単位のデータに分割され、 $4 \times N_k$ ( $N_k=4, 6, 8$ )の長方形行列として表現される。

【0005】そして、AESでは、次のような手順でブロックデータを暗号化する。即ち、図9に示すように、まず、RoundKeyAddition(ラウンドキー加算)と呼ばれる変換を1回行う。次に、ByteSub(バイト置換)、ShiftRow(行シフト)、MixColumn(列ミックス)及び上記RoundKeyAdditionの4種類の変換を順に行うRoundFunction(ラウンド変換)と呼ばれる一連の変換を、複数のラウンド(ラウンド数Nr-1回)に渡って繰り返す。そして最後に、RoundFunctionの中からMixColumnだけを除いたFinalRoundFunction(最終ラウンド変換)と呼ばれる変換を、1回行う。尚、ラウンド数Nrは、下記第1表に示すように、鍵長に対応して決められている。

【0006】

【表1】

第1表

Nk	4	6	8
Nr	10	12	14

【0007】次に、AES暗号方式を用いてデータを暗号化するための上記4種類の変換について、それぞれ説明する。[ByteSub] ByteSubでは、ブロックデータの各単位である各部分データに対して、以下の計算を行う。

【0008】GF(2<sup>8</sup>)における乗法的逆元を計算

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad \cdots \text{式(2)}$$

【0010】このように、ByteSubで行う本来の処理は、上記のようなものであるが、本処理では、入力された1バイト長のデータに対して出力を一に決定できることから、実際には、図10に示すように、1バイトのデータを1バイトのデータに変換するS-boxと呼ばれるルックアップテーブルを先に計算しておき、このルックアップテーブルを用いて、各部分データを変換する。

【0011】つまり、ByteSubでは、引数であるS-boxを用いて、ブロックデータを部分データ単位

する(乗算は多項式の積を「 $x^8 + x^4 + x^3 + x + 1$ 」でmodをとる)。

上記の結果に、各ビットに式(2)のアフィン変換を行う。

10 【0009】

【数6】

(8ビット単位)で変換するようになっている。

[ShiftRow] ShiftRowでは、図11に示すように、ブロックデータにおける1行目から3行目の各行にそれぞれ対応する引数であるC1、C2、C3に基づき、各行のデータをバイト単位で循環左シフトする。これにより、各行の部分データは、下記第2表に示すオフセット値だけ左方向へ循環シフトする。

【0012】

【表2】

第2表

	0行目	1行目	2行目	3行目
オフセット値	0	1	2	3

【0013】つまり、ShiftRowでは、各行に対応する引数Ciを用いて、ブロックデータを行単位(32ビット単位)で変換するようになっている。

[MixColumn] MixColumnでは、ブロックデータにおける各列に対して、GF(2<sup>8</sup>)におい

$$c(x) = '03'x^3 + '01'x^2 + '01'x + '02' \quad \cdots \text{式(3)}$$

そして、得られた多項式の係数を対応する各部分データの値とする。つまり、MixColumnでは、図12に示すように、ブロックデータにおける各列のデータに対して行列Cをかけ算することで変換する(即ち、引数である行列Cを用いて、ブロックデータを列単位(32ビット単位)で変換する)ようになっている。

【0015】[RoundKeyAddition] RoundKeyAdditionでは、引数としてRoundKey(ラウンドキー)と呼ばれる鍵を用いる。

て、各部分データを係数とする多項式と下記の式(3)との乗算を行い、その積を「 $x^4 + 1$ 」でmodをとる。

【0014】

そして、このRoundKeyは、前述した元々の鍵(図8)から生成される。

【0016】即ち、図13に示すように、元々の鍵を拡張アルゴリズムに従って拡張し、その拡張した鍵をブロックデータのブロック長(128ビット)で順番に区切ることにより、図9に示したブロックデータの暗号化手順に含まれるRoundKeyAdditionの回数分(ラウンド数Nr+1回分)のRoundKeyを生成する。こうして生成した複数のRoundKey

は、データの暗号化のために複数回行われるRound Key Additionで順に用いられる。

【0017】そして、RoundKeyAdditionでは、図14に示すように、ブロックデータとRoundKeyとのEXOR（排他論理和）をビット単位でとる。つまり、RoundKeyAdditionでは、引数であるRoundKeyを用いて、ブロックデータを1ビット単位で変換するようになっている。

【0018】以上説明した4種類の変換からなるRoundFunctionの処理のイメージを図15に示す。ところで、こうしたデータの暗号化処理では、処理速度を向上させるための手法として、ルックアップテーブルが一般に用いられる。

【0019】ここで、AESでは、128ビットのブロックデータ毎に変換が行われるため、これを1つのルックアップテーブルを用いて変換しようすると、「 $2^{128} \times 16$  バイト」といった膨大な大きさのルックアップテーブルが必要となってしまう、現実的ではない。

【0020】そこで、本発明者は、AES暗号方式の特徴に着目して、次のようなルックアップテーブルを考えた。即ち、RoundFunctionでは、行単位の変換であるShiftRowと、列単位の変換であるMixColumnとが行われることで、結果として128ビット単位で変換が行われるようになっているが、ShiftRowでは、部分データの位置を移動させているだけであるため、この点を考慮すれば、RoundFunction1回分の処理については、列単位（32ビット単位）の変換であるとみなすことができる。つまり、ルックアップテーブルを用いて変換する32ビットのデータとして、ShiftRowの変換を施すことにより同じ列に移動する4つの部分データを予め選択することで、ShiftRowを考慮する必要がなくなるのである。

【0021】このため、列単位（32ビット単位）で変

$$h_j = T_0[a_{0,j}] \oplus T_1[a_{1,j-C1}] \oplus T_2[a_{2,j-C2}] \oplus T_3[a_{3,j-C3}] \dots \text{式(1)}$$

$$\left( \begin{array}{l} \text{但し、} h_j : \text{変換後データにおける} j \text{ 列目} (j=0, 1, 2, 3) \\ \text{のデータ(32ビット)} \\ T_0 \sim T_3 : 8 \text{ ビットのデータを32ビットのデータ} \\ \text{に変換するルックアップテーブル} \\ a_{i,j} : \text{変換対象データにおける} i \text{ 行目} (i=0, 1, 2, 3) \\ \text{の} j \text{ 列目のデータ(8ビット)} \\ \text{(尚、} a_{i,j-Ci} \text{ は、ShiftRowの変換を施す} \\ \text{ことにより} i \text{ 行目の} j \text{ 列目へ移動するデータ} \\ \text{を表す)} \\ \oplus : \text{排他論理和} \end{array} \right)$$

換を行うルックアップテーブルを用いてデータを変換することが可能となる。

【0022】

【発明が解決しようとする課題】しかしながら、列単位のルックアップテーブルであっても、「 $2^{32} \times 4$  バイト = 16384メガバイト」といった非常に大きなものとなるため、データを暗号化する装置にも非常に大きなデータ記憶容量が必要になってしまう。

【0023】一方また、こうした問題に加え、データの暗号化と復号化との両方を行うためのプログラムでは、暗号化のみを行うためのプログラムに比べ、更に大きなデータ記憶容量が必要となってしまう。AES暗号方式を用いて暗号化されたデータは、暗号化の処理と全く逆の処理を行うことで復号化することができるが、この場合には、暗号化と復号化とで2種類のプログラムが必要となってしまうからである。

【0024】以上のように、暗号化や復号化といったデータ変換処理を行う装置に大きなデータ記憶容量が必要になると、特に携帯電話装置のような小型機器への実装が困難になる。本発明は、こうした問題に鑑みなされたものであり、AES暗号方式を用いてデータを暗号化又は復号化する装置に必要なデータ記憶容量を小さくすることを目的としている。

【0025】

【課題を解決するための手段及び発明の効果】上記目的を達成するためになされた請求項1に記載のデータ変換装置は、4行4列の行列で表される128ビットの変換対象データに、AES暗号方式におけるByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られる変換後データを、下記式(1)により求めるように構成されている。

【0026】

【数7】

【0027】つまり、請求項1のデータ変換装置では、50 変換対象データにByteSub、ShiftRow及

びMixColumnの各変換を順に施すことで得られる変換後データを上記式(1)により求める、といった請求項4のデータ変換方法を用いている。

【0028】ここで、各ルックアップテーブル $T_0 \sim T_3$ の大きさは、それぞれ「 $2^8 \times 4$ バイト=1024バイト」であるから、4つのルックアップテーブルを合わせても4096バイトである。この値は、前述した列単位で変換を行うルックアップテーブルの大きさ(16384メガバイト)に比べ、格段に小さい。

【0029】このような請求項1のデータ変換装置によれば、AES暗号方式を用いたデータの暗号化の処理速度を向上させつつ、AES暗号方式を用いてデータを暗号化する装置に必要なデータ記憶容量を小さくすることができる。即ち、AES暗号方式を用いたデータの暗号化では、ByteSub、ShiftRow及びMixColumnの各変換を順に行う処理を複数回繰り返すようになっているが、この処理で得られる値を本データ変換装置(請求項4のデータ変換方法)により求めることで、各変換を順に行う必要がなくなり、処理速度を向上させることができる。しかも、使用するルックアップテーブルが小さいことから、AES暗号方式を用いてデータを暗号化する装置に必要なデータ記憶容量を小さくすることができる。

【0030】次に、請求項2に記載のデータ変換プログラムは、4行4列の行列で表される128ビットの変換対象データに、AES暗号方式におけるByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られる変換後データを、上記式

(1)により求める機能をコンピュータに実現させるためのものである。

【0031】つまり、請求項2のデータ変換プログラムでも、変換対象データにByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られる変換後データを、請求項4のデータ変換方法を用いて求めるようになっている。

【0032】このような請求項2のデータ変換プログラムを、AES暗号方式の暗号化を行うために用いれば、データの暗号化の処理速度を向上させつつ、AES暗号方式を用いてデータを暗号化する装置に必要なデータ記憶容量を小さくすることができる。また、請求項2のデータ変換プログラムは、請求項3のようにコンピュータ読み取り可能な記録媒体に記録されていてもよい。

【0033】次に、請求項5に記載のデータ変換方法は、AES暗号方式を用いて暗号化されたデータを、下記(a1)～(a4)の条件に従い復号化することの特徴としている。

(a1): ByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換を、AES暗号方式の暗号化手順と同じ順序で行う。

【0034】(a2): 上記(a1)に従い行うByteSub、ShiftRow及びMixColumnでは、暗号化で行われる変換の逆変換となるような引数を用いる。

(a3): 上記(a1)に従い行うRoundKeyAdditionでは、暗号化で用いられた複数のRoundKeyを、暗号化と逆の順序で用いる。

【0035】(a4): 更に、上記(a1)に従い行うRoundKeyAdditionのうち、MixColumnの次に行うRoundKeyAdditionでは、上記(a3)に従い用いるRoundKeyに、復号化で行うMixColumnの変換を施したものを、RoundKeyとして用いる。

【0036】つまり、請求項5のデータ変換方法では、ByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換に、暗号化の場合とは異なる引数を用いることで、AES暗号方式の暗号化手順と同じ順序で各変換を行いつつデータを復号化するようになっている。

【0037】このような請求項5のデータ変換方法を、AES暗号方式の暗号化及び復号化を行うためのプログラムに用いれば、そのプログラムサイズを小さくすることができる。つまり、ByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換を暗号化手順と同じ順序で行うためのプログラムを共用できるからである。

【0038】次に、上記請求項5のデータ変換方法を用いた請求項7に記載のデータ変換プログラムは、AES暗号方式を用いたデータの暗号化と、AES暗号方式を用いて暗号化されたデータの復号化とを、コンピュータに行わせるためのものである。

【0039】そして、このデータ変換プログラムは、データの暗号化及び復号化で共通に用いられると共に、そのデータに対してByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換をAES暗号方式の暗号化手順と同じ順序で行うための共通プログラムと、この共通プログラムでの各変換に用いられる引数を、暗号化の場合にはAES暗号方式の暗号化手順において用いられる引数に設定し、復号化の場合には下記(b1)～(b3)の条件に従い設定する設定プログラムと、を備えている。

【0040】(b1): ByteSub、ShiftRow及びMixColumnでは、暗号化で行われる変換の逆変換となるような引数が用いられるようにする。

(b2): RoundKeyAdditionで用いられる引数であるRoundKeyとしては、暗号化で用いられた複数のRoundKeyが、暗号化と逆の順序で用いられるようにする。

【0041】(b3): 更に、MixColumnの次に行われるRoundKeyAdditionで用いら

れる引数であるRoundKeyとしては、上記(b2)に従い用いられるRoundKeyに、復号化で行われるMixColumnの変換を施したものが用いられるようにする。

【0042】このような請求項7のデータ変換プログラムは、データを暗号化するためのプログラムとデータを復号化するためのプログラムとを独立して備えるものに比べ、プログラムサイズが格段に小さい。したがって、このデータ変換プログラムによれば、AES暗号方式を用いてデータの暗号化及び復号化を行う装置に必要なデータ記憶容量を小さくすることができる。

【0043】ところで、請求項4のデータ変換方法によれば、変換対象データにByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られる変換後データを上記式(1)により求めることで、データの暗号化の処理速度を向上させつつ、AES暗号方式を用いてデータを暗号化する装置に必要なデータ記憶容量を小さくすることができる。一方、上記請求項5のデータ変換方法ではデータを復号化する場合に、また上記請求項7のデータ変換プログラムではデータを復号化する場合にも、AES暗号方式の暗号化手順と同じ順序でByteSub、ShiftRow、MixColumnの各変換を行うこととなる。よって、上記請求項4のデータ変換方法は、請求項6又は請求項8に記載のように、請求項5と請求項7とのそれぞれにも適用することができる。

【0044】即ち、請求項6に記載のデータ変換方法では、上記請求項5のデータ変換方法において、4行4列の行列で表される128ビットの変換対象データにByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られる変換後データを、上記式(1)により求めることを特徴としている。

【0045】また、請求項8に記載のデータ変換プログラムでは、上記請求項7のデータ変換プログラムにおいて、共通プログラムが、4行4列の行列で表される128ビットの変換対象データにByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られる変換後データを、上記式(1)により求めるようになっている。

【0046】そして、上記請求項6のデータ変換方法によれば、AES暗号方式を用いて暗号化されたデータの復号化の処理速度を向上させつつ、AES暗号方式を用いてデータを復号化する装置に必要なデータ記憶容量を小さくすることができる。更に、データの暗号化と復号化との両方をコンピュータに行わせるためのデータ変換プログラムに適用すれば、請求項5の効果に加え、データの暗号化と復号化との両方の処理速度を向上させることができる。

【0047】また、上記請求項8のデータ変換プログラムによれば、請求項7の効果に加え、データの暗号化及

び復号化の処理速度を向上させつつ、AES暗号方式を用いてデータの暗号化及び復号化を行う装置に必要なデータ記憶容量を小さくすることができる。

【0048】尚、請求項7、8のデータ変換プログラムは、請求項9のようにコンピュータ読み取り可能な記録媒体に記録されていてもよい。

【0049】

【発明の実施の形態】以下、本発明が適用された実施形態のデータ変換装置としての携帯電話装置について、図面を用いて説明する。まず図1は、本実施形態の携帯電話装置10の構成を表すブロック図である。

【0050】この携帯電話装置10は、送受信部12、CPU14、RAM16、ROM18及びデータの書き換えが可能な不揮発性メモリ20を備えている。送受信部12は、無線通信によりデータの送受信を行う。また、不揮発性メモリ20には、外部のデータベースとの間でデータをやりとりするデータ送受信処理をCPU14に行わせるためのアプリケーションプログラムと、このデータ送受信処理で送受信されるデータをAES暗号方式を用いて暗号化又は復号化する処理(以下、「データ変換処理という」)をCPU14に行わせるためのデータ変換プログラムとが記憶されている。

【0051】ここで、上記データ変換プログラムに従いCPU14が行うデータ変換処理について、図2のフローチャートを用いて説明する。尚、本データ変換処理は、上記データ送受信処理にて平文データ(即ち、暗号化されていないデータ)を外部のデータベースへ送信しようとする場合又は外部のデータベースから暗号データを受信した場合に開始される。

【0052】このデータ変換処理が開始されると、まずS100にて、暗号化と復号化との何れの処理を行うのかを判定する。具体的には、上記データ送受信処理にて平文データを送信しようとする場合には、暗号化を行うと判定し、上記データ送受信処理にて暗号データを受信した場合には、復号化を行うと判定する。

【0053】そして、S100で、暗号化を行うと判定した場合には、S110へ移行し、暗号化用のRoundKeyを生成して、RAM16に記憶させる。ここで、暗号化用のRoundKeyとは、平文データを暗号化する際に行うRoundKeyAdditionで用いるRoundKeyのことであり、従来技術(図13)で説明したように、鍵を拡張して順番に区切ることでにより複数生成する。

【0054】このS110で暗号化用のRoundKeyを生成した後は、S120へ移行し、後述するByteSub、ShiftRow及びRoundKeyAdditionの各変換に用いる引数を、暗号化用の引数に設定する。具体的には、従来技術で説明したように、ByteSubで用いる引数を、図10で説明したS-boxに設定する。また、ShiftRowで用いる引



数を、図11で説明したCiに設定する。更に、RoundKeyAdditionで用いる引数を、図14で説明したRoundKey（即ち、S110にて生成した暗号化用のRoundKey）に設定する。

【0055】そして次に、S130へ移行し、後述するRoundFunctionで用いる暗号化用の4つのルックアップテーブルT<sub>0</sub>~T<sub>3</sub>を生成して、RAM16に記憶させる。一方、S100で、復号化を行うと判定した場合には、S140へ移行し、復号化用のRoundKeyを生成して、RAM16に記憶させる。ここで、復号化用のRoundKeyとは、暗号データを復号化する際に行うRoundKeyAdditionで用いるRoundKeyのことであり、次の(c1)、(c2)の条件を満たすものである。

【0056】(c1): 復号化で行うRoundKeyAdditionでは、暗号化で用いられた複数のRoundKeyを、暗号化と逆の順序で用いる。

(c2): 但し、(c1)の条件に従い用いるRoundKeyのうち、RoundFunctionで行うRoundKeyAdditionで用いるRoundKeyについては、更に、暗号化で行うMixColumnの逆変換を施したものをRoundKeyとして用いる。

【0057】したがって、復号化用のRoundKeyは、次のように生成される。まず、暗号化の場合と同様に、鍵を拡張して順番に区切ることにより複数のRoundKey（即ち、暗号化で用いられるRoundKey）を生成する。次に、生成した複数のRoundKeyを、暗号化とは逆の順序で各RoundKeyAdditionに割り振る。更に、RoundFunctionで用いるRoundKeyについては、暗号化で行うMixColumnの逆変換を施す。

【0058】こうしてS140で復号化用のRoundKeyを生成した後は、S150へ移行し、後述するByteSub、ShiftRow及びRoundKeyAdditionの各変換に用いる引数を、復号化用の引数に設定する。具体的には、ByteSubで用いる

引数を、暗号化で用いるS-boxの逆変換となるルックアップテーブルに設定する。また、ShiftRowで用いる引数を、暗号化で用いるCiの逆変換となる引数（即ち、ブロックデータにおける各行の部分データを暗号化の場合と同じオフセット値だけ右方向へ循環シフトする引数）に設定する。更に、RoundKeyAdditionで用いる引数を、S140にて生成した復号化用のRoundKeyに設定する。

【0059】そして次に、S160へ移行し、後述するRoundFunctionで用いる復号化用の4つのルックアップテーブルT<sub>0</sub>~T<sub>3</sub>を生成して、RAM16に記憶させる。こうして、S130又はS160でルックアップテーブルT<sub>0</sub>~T<sub>3</sub>を生成した後は、以下のS170~S220にて、AES暗号方式の暗号化手順と同じ順序でByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換を行うようになっている。尚、S100~S160の処理をCPU14に行わせるためのプログラムが、設定プログラムに相当し、S170~S220の処理をCPU14に行わせるためのプログラムが、共通プログラムに相当する。

【0060】まず、S170では、暗号データ又は平文データであるブロックデータに対して、RoundKeyAdditionの変換を1回行う。次に、S180へ移行し、RoundFunctionを行う。ここで、本データ変換処理におけるRoundFunctionの処理について説明する。

【0061】RoundFunctionは、ByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換を順に施す処理であるが、このS180では、各変換をそれぞれ行うのではなく、4行4列の行列で表される128ビットの処理対象データ（変換対象データに相当）にRoundFunctionを行うことで得られる処理後データを、下記式(4)により求めるようになっている。

【0062】

【数8】

$$e_j = T_0[a_{0,j}] \oplus T_1[a_{1,j-c_1}] \oplus T_2[a_{2,j-c_2}] \oplus T_3[a_{3,j-c_3}] \oplus k_j \cdots \text{式(4)}$$

$$\left( \begin{array}{l} \text{但し、} e_j : \text{処理後データにおける} j \text{列目} (j=0, 1, 2, 3) \\ \text{のデータ(32ビット)} \\ T_0 \sim T_3 : 8 \text{ビットのデータを32ビットのデータ} \\ \text{に変換するルックアップテーブル} \\ a_{i,j} : \text{処理対象データにおける} i \text{行目} (i=0, 1, 2, 3) \\ \text{の} j \text{列目のデータ(8ビット)} \\ \text{(尚、} a_{1,j-c_i} \text{は、ShiftRowの変換を施す} \\ \text{ことにより} i \text{行目の} j \text{列目へ移動するデータ} \\ \text{を表す)} \\ k_j : \text{RoundKeyにおける} j \text{列目} (j=0, 1, 2, 3) \\ \text{のデータ(32ビット)} \\ \oplus : \text{排他論理和} \end{array} \right)$$

【0063】上記式(4)では、処理対象データから処理後データへの変換を32ビット単位で行う。具体的には、図3に示すように、処理対象データを構成する部分データ(8ビット)の中から、ShiftRowの変換を施すことにより同列に移動する4つの部分データ( $a_{0,j}$ 、 $a_{1,j-c_1}$ 、 $a_{2,j-c_2}$ 、 $a_{3,j-c_3}$ )を選択し、この選択した4つの部分データからなる32ビットのデータを、処理後データにおける同列の4つの部分データからなる32ビットのデータ $e_j$ に変換するようになっている。

【0064】また、上記式(4)では、処理対象データにByteSub、ShiftRow及びMixColumnの各変換を順に施すことで得られるデータ(変換後データに相当)を、4つのルックアップテーブル $T_0 \sim T_3$ を用いて求めるようになっている。具体的には、処理対象データから選択した各部分データ( $a_{0,j}$ 、 $a_{1,j-c_1}$ 、 $a_{2,j-c_2}$ 、 $a_{3,j-c_3}$ )を、対応するルックアップテーブル $T_0 \sim T_3$ を用いてそれぞれ変換し、この変換後の値の排他論理和をとることで求めている。そして更に、このように求めたデータに、RoundKeyAdditionの変換(即ち、上記式(4)における $k_j$ の加算)を施して、処理後データを求めるようになっている。尚、このRoundKeyAdditionの変換が、MixColumnの次に行うRoundKeyAdditionに相当する。

【0065】ここで、各ルックアップテーブル $T_0 \sim T_3$ は、処理対象データの部分データ(8ビット)を32ビットのデータに変換するものである。したがって、各ルックアップテーブル $T_0 \sim T_3$ の大きさは、それぞれ「 $2^8 \times 4 \text{バイト} = 1024 \text{バイト}$ 」であり、4つのルックアップテーブルを合わせても4096バイトである。

【0066】そして、このようなS180の処理を行った後はS190へ移行し、S180の処理を $Nr-1$ 回

行ったか否かを判定する。尚、ラウンド数 $Nr$ は、前述したように、第1表に示す値である。このS190で、 $Nr-1$ 回行っていないと判定した場合には、S180へ戻る。つまり、S180の処理を $Nr-1$ 回繰り返すようになっている。

【0067】一方、S190で、 $Nr-1$ 回行ったと判定した場合には、S200へ移行してByteSubの変換を行い、次に、S210へ移行してShiftRowの変換を行い、更に、S220へ移行してRoundKeyAdditionの変換を行い、本データ変換処理を終了する。

【0068】以上説明したように、上記データ変換処理では、平文データを暗号化する場合だけでなく、暗号データを復号化する場合にも、AES暗号方式の暗号化手順と同じ順序でByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換を行うようになっている。ここで、暗号データの復号化を平文データの暗号化と同じ変換順序で行うことができる理由について説明する。

【0069】そもそもAES暗号方式を用いて暗号化された暗号データは、暗号化の処理と全く逆の処理を行う(即ち、暗号化で行う各変換の逆変換を暗号化とは逆の順序で行う)ことで復号化することができる。RoundFunctionを例にして説明すると、図4に示すように、RoundKeyAddition→InvMixColumn→InvShiftRow→InvByteSubの順に変換を行えばよい。

【0070】ここで、InvMixColumn、InvShiftRow、InvByteSubは、それぞれ、暗号化の場合のMixColumn、ShiftRow、ByteSubの逆変換であり、変換に用いる引数異なるだけである。即ち、InvMixColumnでは、暗号化で用いる行列Cの逆変換となる行列を引

20

30

40

50

数として用いてデータを変換する。また、InvShiftRowでは、暗号化で用いるCiの逆変換となる引数を用いてデータを変換する。また更に、InvByteSubでは、暗号化で用いるS-boxの逆変換となるルックアップテーブルを引数として用いてデータを変換する。尚、RoundKeyAdditionについては、RoundKeyの排他論理和をとる変換であるため、暗号化と同じRoundKeyを用いて変換すれば、逆変換となる。

【0071】そして、InvShiftRow→InvByteSubという変換順序は、InvByteSub→InvShiftRowという順序に入れ替えても、結果は変わらない。また、RoundKeyAddition→InvMixColumnという変換順序は、RoundKeyAdditionで用いるRoundKeyにInvMixColumnの変換を施せば、InvMixColumn→RoundKeyAdditionという順序に入れ替えても、結果は変わらない。

【0072】即ち、RoundKeyAdditionで用いるRoundKeyを「K」、InvMixColumnの変換行列を「D」、変換の対象となるデータを「B」とすると、RoundKeyAddition→InvMixColumnという復号化処理は、下記式(5)の左辺のように表すことができ、更に、右辺のように展開することができる。

$$\text{【0073】 } (B+K) \cdot D = B \cdot D + K \cdot D \quad \cdots \text{式 (5)}$$

そのため、RoundKeyAdditionで、RoundKeyにInvMixColumnの変換を施して得られる値「K・D」をRoundKeyとして用いれば、InvMixColumn→RoundKeyAdditionという順序に入れ替えることができる。

【0074】その結果、図5の左側に示すような復号化の変換順序（即ち、暗号化手順と逆の順序）を、InvShiftRowとInvByteSubとの順序を入れ替え、更に、RoundKeyAdditionとInvMixColumnとの順序を入れ替えることで、図5の右側に示すように、暗号化手順と同じ順序に変更することができる。

【0075】以上の理由から、各変換に用いる引数を、下記(d1)～(d3)の条件に従い設定すれば、暗号化手順と同じ変換順序で復号化を行うことができる。

(d1) : ByteSub、ShiftRow及びMixColumnでは、暗号化で行われる変換の逆変換となるような引数を用いる。

【0076】(d2) : RoundKeyAdditionで用いるRoundKeyとしては、暗号化で用いられた複数のRoundKeyを、暗号化と逆の順序で用いる。

(d3) : 更に、MixColumnの次に行うRoundKeyAdditionで用いるRoundKeyとしては、上記(d2)に従い用いるRoundKeyに、復号化で行うMixColumnの変換を施したものを用いる。

【0077】そして、前述したデータ変換処理では、こうした順序変更に加え、更に、RoundFunctionを上記式(4)により行うようになっている。即ち、暗号化の場合には、処理対象データに対して暗号化用の引数を用いたByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換を順に施すことで得られる処理後データを、式(4)により求めるようになっており、言い換えれば、このような処理後データが得られるような暗号化用のルックアップテーブルT<sub>0</sub>～T<sub>3</sub>をS130にて生成するようになっている。

【0078】また、復号化の場合には、処理対象データに対して復号化用の引数（即ち、上記(d1)～(d3)の条件を満たす引数）を用いたByteSub、ShiftRow、MixColumn及びRoundKeyAdditionの各変換を順に施すことで得られる処理後データを、式(4)により求めるようになっており、言い換えれば、このような処理後データが得られるような復号化用のルックアップテーブルT<sub>0</sub>～T<sub>3</sub>をS160にて生成するようになっている。

【0079】次に、本発明者が行った試験内容について簡単に説明する。本発明者は、AES暗号方式を用いてデータの暗号化及び復号化を行う上記実施形態のデータ変換プログラムを市販の携帯電話装置へ実装し、動作の確認を行った。

【0080】今回、実装の対象とした携帯電話装置は、Java（登録商標）アプリケーションを実行可能な機種であり、Java（登録商標）アプリケーションの大きさについて、Jarファイル（Java（登録商標）の圧縮ファイル）で10キロバイト以内という制約事項があった。本発明者は、実際にAES暗号クラス（上記実施形態のデータ変換プログラム）を作成して、その作成したAES暗号クラスをJarファイルに圧縮した。その結果、暗号クラス（圧縮後のプログラム）の大きさは2.4キロバイトとなり、この携帯電話装置に実装することができた。

【0081】そして、図6に示すように、この携帯電話装置から外部の就職先データベースへアクセスするアプリケーションを開発し、送信データの暗号化及び受信データの復号化を実際に行えることを確認した。このような本実施形態の携帯電話装置10によれば、AES暗号方式を用いたデータの暗号化及び復号化を、速い処理速度で、しかも比較的小さなデータ記憶容量で実現することができる。

【0082】ここで、暗号化及び復号化の処理速度が速

くなる理由としては、データ変換処理において、処理対象データにRoundFunctionの処理を行うことで得られる処理後データを上記式(4)により求めるといったデータ変換方法が用いられていることが挙げられる。

【0083】一方、データ記憶容量を小さくすることができる理由としては、データ変換処理をCPU14に行わせるためのデータ変換プログラムのプログラムサイズが小さいことが挙げられる。即ち、データ変換処理において、暗号化と復号化とを同一の変換順序で行うといったデータ変換方法が用いられており、しかも、上記式(4)で使用するルックアップテーブルT<sub>0</sub>~T<sub>3</sub>を、データ変換処理の中で生成するようになっているからである。

【0084】また、データ記憶容量が小さくできるもう一つの理由としては、上記式(4)で使用するルックアップテーブルT<sub>0</sub>~T<sub>3</sub>が小さいことが挙げられる。以上、本発明の一実施形態について説明したが、本発明は、種々の形態を採り得ることは言うまでもない。

【0085】例えば、上記実施形態の携帯電話装置10では、データ変換処理を開始してからルックアップテーブルT<sub>0</sub>~T<sub>3</sub>を生成するようになっているが、これに限らず、データ変換プログラムの一部としてルックアップテーブルT<sub>0</sub>~T<sub>3</sub>のデータを予め記憶していてもよい。このようにすれば、ルックアップテーブルT<sub>0</sub>~T<sub>3</sub>を生成する必要がなくなる分、処理速度をより速くすることができる。

【0086】また、上記実施形態の携帯電話装置10では、データを暗号化又は復号化する際に行うRoundKeyAdditionで用いる複数のRoundKeyを一度に生成するようになっているが、これに限らず、RoundKeyAdditionを行う毎に、そこで必要となるRoundKeyのみを生成するようにし、不要になったRoundKeyについてはすぐに消去するようにしてもよい。このようにすれば、RAM16に必要なデータ容量をより小さくすることができる。

【0087】一方、上記実施形態では、データの暗号化及び復号化を行うデータ変換装置としての携帯電話装置10について述べたが、データ変換プログラム自体や、

このデータ変換プログラムが記録されたコンピュータ読み取り可能な記録媒体(例えば、フロッピー(登録商標)ディスクやCD-ROM等)や、データ変換方法自体も、本発明の範囲である。

【図面の簡単な説明】

【図1】 実施形態の携帯電話装置の構成を表すブロック図である。

【図2】 データ変換処理を表すフローチャートである。

10 【図3】 データ変換処理におけるRoundFunctionの処理のイメージを表す説明図である。

【図4】 データを復号化する場合のRoundFunctionの処理のイメージを表す説明図である。

【図5】 データを復号化する場合の変換順序を表す説明図である。

【図6】 データ変換プログラムを市販の携帯電話装置へ実装した結果を説明する説明図である。

【図7】 AES暗号方式に用いられるブロックデータを説明する説明図である。

20 【図8】 AES暗号方式に用いられる鍵を説明する説明図である。

【図9】 AES暗号方式の暗号化手順を説明する説明図である。

【図10】 ByteSubの変換を説明する説明図である。

【図11】 ShiftRowの変換を説明する説明図である。

【図12】 MixColumnの変換を説明する説明図である。

30 【図13】 RoundKeyの生成方法を説明する説明図である。

【図14】 RoundKeyAdditionの変換を説明する説明図である。

【図15】 RoundFunctionの処理のイメージを表す説明図である。

【符号の説明】

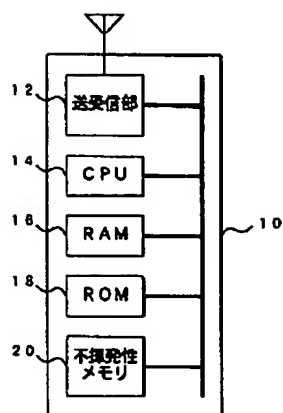
10…携帯電話装置、12…送受信部、14…CPU、16…RAM、18…ROM、20…不揮発性メモリ

【図7】

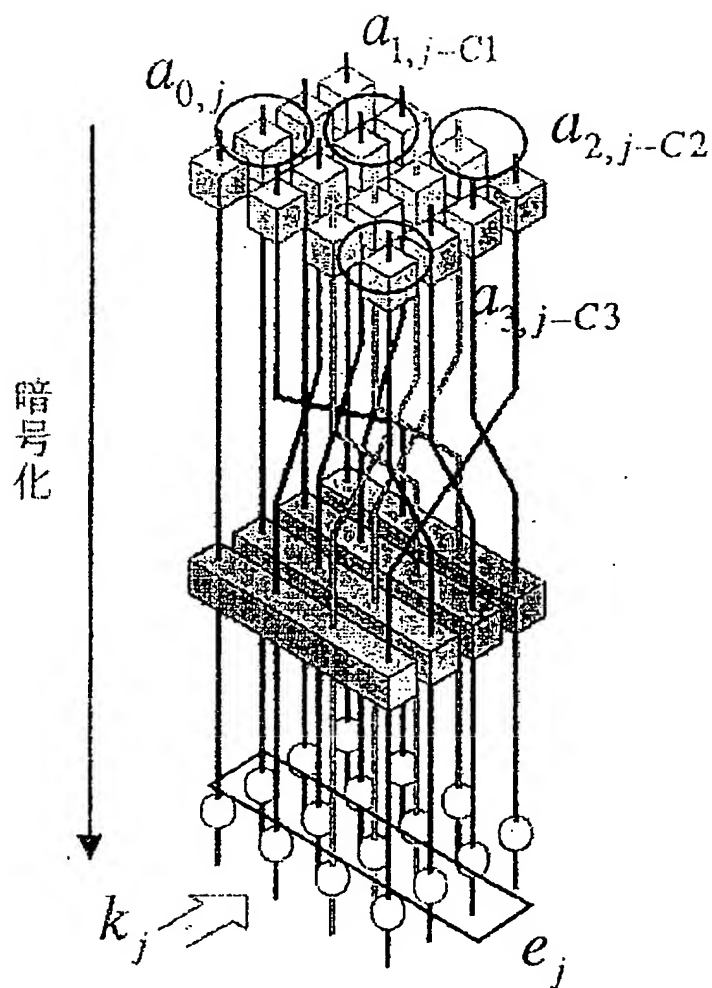
ブロックサイズ:  
128bit



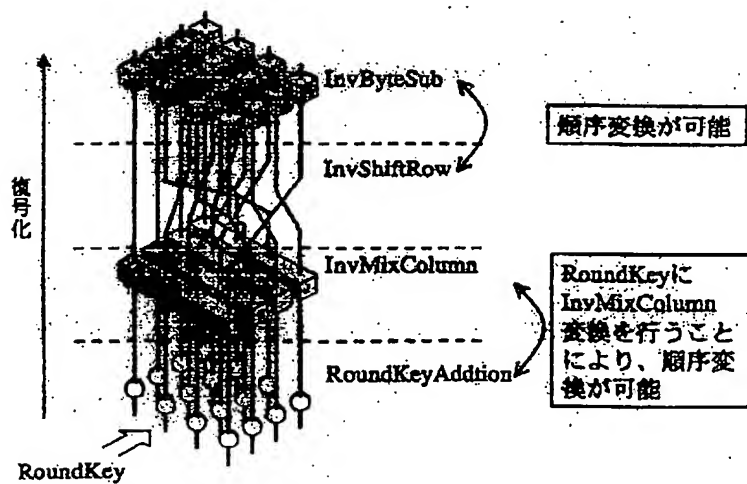
【図1】



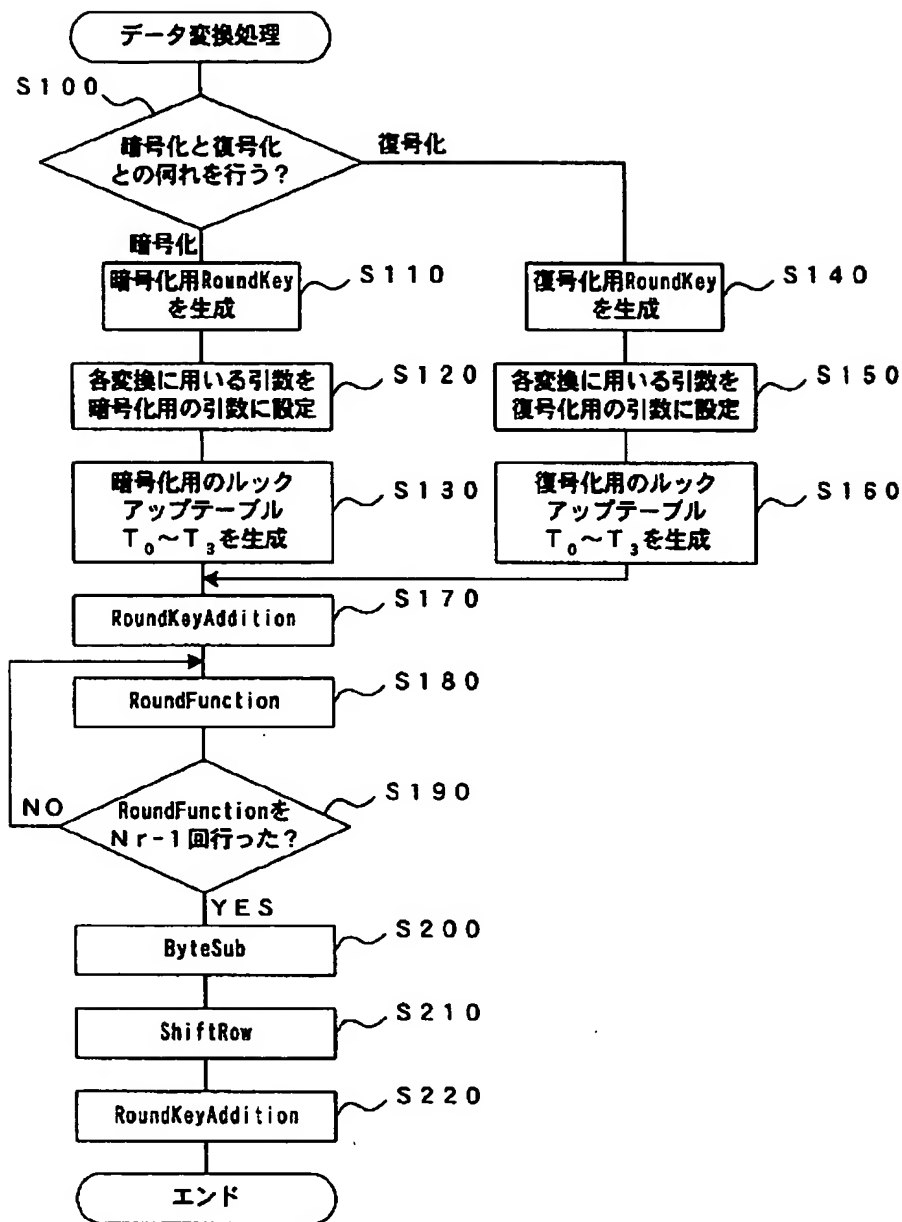
【図3】



【図4】



【図2】



【図8】

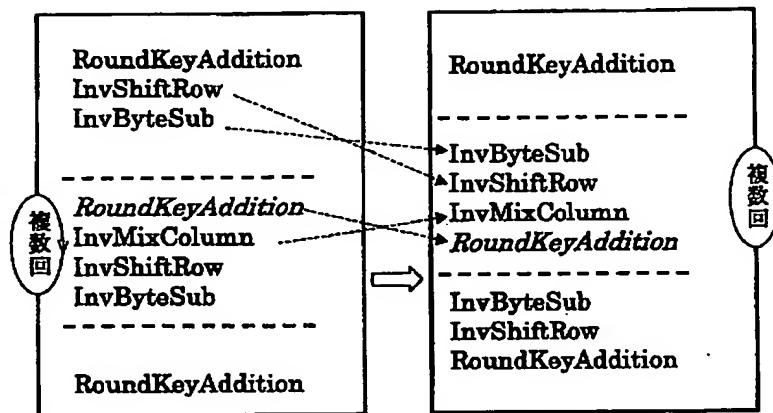
鍵サイズ:  
128, 192, 256bit

128



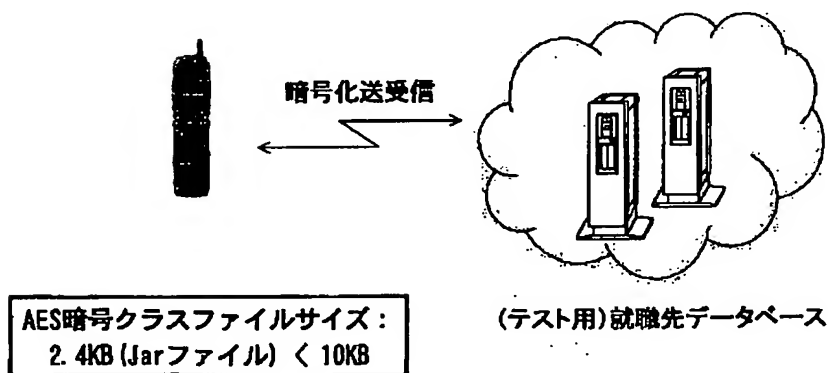
105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000

【図5】



【図6】

## 実装結果



**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**